RW

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/802,111	03/07/2001	Todor J. Fay	MS1-737US	5579

22801 7590 05/03/2005

LEE & HAYES PLLC
421 W RIVERSIDE AVENUE SUITE 500
SPOKANE, WA 99201

EXAMINER	
FLANDERS, ANDREW C	

ART UNIT	PAPER NUMBER
2644	

DATE MAILED: 05/03/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	09/802,111	FAY ET AL.
	Examiner	Art Unit
	Andrew C Flanders	2644

-- *The MAILING DATE of this communication appears on the cover sheet with the correspondence address* --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *24 March 2005*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-56* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-56* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *07 March 2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____.

DETAILED ACTION

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. **Claims 1, 3 – 21, 36, 37, 41 – 45, 47, 48, and 50 – 56** are rejected under 35 U.S.C. 103(a) as being unpatentable over Tulkoff (U.S. Patent 5,890,017) in view of Solomon (U.S. Patent 5,768,545).

Regarding **Claim 1**, Tulkoff discloses a Mixer that receives multiple audio streams from multiple audio processes and an audio device (Fig. 1 elements 10, 12 and 14) (i.e. receiving multiple streams of audio wave data and an audio wave data consumer), the audio processes make connections to the mixer (col. 3 lines 17 – 18) (i.e. defining logical buses that each correspond to an audio wave data consumer), audio streams are sent from the audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12). It is inherent that if the streams are sent from the audio process to the mixer that some sort of connection device or bus must be present to transport the streams (i.e. assigning at least one of the multiple streams of audio wave data to a plurality of the logical buses; and routing any audio wave data stream assigned to a particular logical bus to the audio wave data consumer corresponding to said particular logical bus). Tulkoff does not disclose dynamically defining the plurality of buses in response to a need associated with receiving the

streams of audio wave data or dynamically de-allocating at lest one of the logical buses when no longer needed to route a stream of audio wave data. Solomon discloses a system with multiple buses, when a sound card is granted access to a bus, the bus locks out all other agents and all other buses until the transaction for which the sound card was granted access has been entirely completed (col. 3 lines 25 – 38) (i.e. dynamically defining a plurality of logical buses in response to a need associated with receiving the streams of audio wave data, the logical buses each corresponding to an audio wave data consumer and dynamically de-allocating at least one of the logical buses when no longer needed to route a stream of audio wave data). It would have been obvious to one of ordinary skill in the art at the time of the invention to use Solomon's bus allocation method on Tulkoff's mixer to control the buses used by Tulkoff's audio processes. One would have been motivated to do so to ensure the audio device would be guaranteed allocation of the bus for the time it is needed. See Solomon col. 1 lines 1 – 45.

Regarding **Claim 3** in addition to the elements stated above regarding claim 1, Tulkoff further discloses the audio processes make connections to the mixer (col. 3 lines 17 – 18) and audio streams are sent from the audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12) (i.e. wherein a plurality of audio wave data streams are assigned to at least one of the logical buses).

Regarding **Claim 4** in addition to the elements stated above regarding claim 1, Tulkoff further discloses a single mixer (audio wave data consumer) that receives audio streams from an audio process and each process sends a single stream to the mixer

(Fig. 1 elements 10 and 12) (i.e. wherein each logical bus corresponds to a single audio wave data consumer).

Regarding **Claim 5**, in addition to the elements stated above regarding claim 1, Tulkoff further discloses that multiple audio streams from audio processes are sent to a single mixer (Fig. 1 elements 10 and 12) (i.e. wherein at least two of the logical buses correspond to the same audio wave data consumer).

Regarding **Claim 6**, in addition to the elements stated above regarding claim 1, Tulkoff further discloses that the data being played is copied to a buffer allotted to the client in the mixer (col. 4 lines 24 –25) (i.e. wherein the audio wave data consumer is a data buffer that performs an action of buffering audio wave data prior to outputting the audio wave data).

Regarding **Claim 7**, in addition to the elements stated above regarding claim 1, Tulkoff further discloses a mixer to digitally mix any number of independent audio streams from any number of applications and sends the new, mixed stream to an audio device (col. 2 lines 63 – 66) (i.e. wherein the audio wave data consumer performs an action of effects-processing the audio wave data prior to outputting the audio wave data).

Regarding **Claim 8**, in addition to the elements stated above regarding claim 1, Tulkoff further discloses that the internal client structures suitably contain all of the audio stream parameters that were set in each audio process (col. 3 lines 19 – 21) (i.e. creating a data structure), the audio processes make connections to the mixer (col. 3 lines 17 – 18) and audio streams are sent from the audio processes to the audio device

by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12) (i.e. correlating the logical busses with corresponding audio wave data consumers).

Regarding **Claim 9**, in addition to the elements stated above regarding claim 1, Tulkoff further discloses that the internal client structures suitably contain all of the audio stream parameters that were set in each audio process (col. 3 lines 19 – 21) (i.e. creating a data structure), the audio processes make connections to the mixer (col. 3 lines 17 – 18) and audio streams are sent from the audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12) (i.e. correlating the logical busses with corresponding audio wave data consumers), the mixer waits on a select system call for client events and the mixer keeps track of all clients that have made connections to it (col. 3 lines 15 – 18) (i.e. referring to the data structure).

Regarding **Claim 10**, in addition to the elements stated above regarding claim 1, Tulkoff further discloses in a running process on a computer a method that includes sending a plurality of audio streams of varying formats from at least one application program running on the computer system to the output audio device and the method further includes intercepting the plurality of audio streams with a mixer daemon for maintaining transparency to the at least one application program (col. 2 lines 1 – 9) (i.e. instantiating a programming object to receive the multiple streams of audio wave data).

Regarding **Claim 11**, in addition to the elements stated above regarding claim 1, Tulkoff further discloses in a running process on a computer a method that includes sending a plurality of audio streams of varying formats from at least one application program running on the computer system to the output audio device and the method

further includes intercepting the plurality of audio streams with a mixer daemon for maintaining transparency to the at least one application program (col. 2 lines 1 – 9) (i.e. programming object to receive the multiple streams of audio wave data, and wherein said routing comprises calling an interface of the programming object).

Regarding **Claim 12**, in addition to the elements stated above regarding claim 1, Tulkoff further discloses providing a running process in a computer system that allows simultaneous use of an output audio device (col. 2 lines 1 – 3) (i.e. one or more computer-readable media comprising computer executable instructions that, when executed, direct a computing system to perform the method of claim 1).

Regarding **Claim 13**, Tulkoff discloses a Mixer and an audio device that receive multiple audio streams from audio processes (Fig. 1 elements 10, 12 and 14) (i.e. a plurality of audio wave data sources that produce one or more streams of audio wave data and a plurality of audio wave data consumers that receive one or more streams of audio wave data) and in a running process on a computer a method that includes sending a plurality of audio streams of varying formats from at least one application program running on the computer system to the output audio device and the method further includes intercepting the plurality of audio streams with a mixer daemon for maintaining transparency to the at least one application program (col. 2 lines 1 – 9) (i.e. a software component that defines logical buses corresponding respectively to the plurality of audio wave data consumers, and he software component configured to receive one or more of the streams of audio wave data at each of the defined logical buses, and route any audio wave data that is received at a particular logical bus to an

audio wave data consumer corresponding to said particular logical bus). Tulkoff does not disclose that the software component dynamically defines logical buses in response to a need associated with receiving the streams of audio wave data and that de-allocates at least one of the logical buses when no longer needed. Solomon discloses a system with multiple buses, when a sound card is granted access to a bus, the bus locks out all other agents and all other buses until the transaction for which the sound card was granted access has been entirely completed (col. 3 lines 25 – 38) (i.e. that the software component dynamically defines logical buses in response to a need associated with receiving the streams of audio wave data and that de-allocates at least one of the logical buses when no longer needed). It would have been obvious to one of ordinary skill in the art at the time of the invention to use Solomon's bus allocation method on Tulkoff's mixer to control the busses used by Tulkoff's audio processes. One would have been motivated to do so to ensure the audio device would be guaranteed allocation of the bus for the time it is needed. See Solomon col. 1 lines 1 – 45.

Regarding **Claim 14**, in addition to the elements listed above regarding claim 13, Tulkoff further discloses a single mixer (audio wave data consumer) that receives audio streams from an audio process and each process sends a single stream to the mixer (Fig. 1 elements 10 and 12) (i.e. wherein each logical bus corresponds to a single audio wave data consumer).

Regarding **Claim 15**, in addition to the elements stated above regarding claim 13, Tulkoff further discloses that multiple audio streams from audio processes are sent

to a single mixer (Fig. 1 elements 10 and 12) (i.e. wherein at least two of the logical buses correspond to the same audio wave data consumer).

Regarding **Claim 16** in addition to the elements stated above regarding claim 13, Tulkoff further discloses the audio processes make connections to the mixer (col. 3 lines 17 – 18) and audio streams are sent from the audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12) (i.e. wherein a plurality of audio wave data streams are assigned to at least one of the logical buses).

Regarding **Claim 17**, in addition to the elements stated above regarding claim 13, Tulkoff further discloses that the data being played is copied to a buffer allotted to the client in the mixer (col. 4 lines 24 –25) (i.e. wherein the audio wave data consumer is a data buffer that performs an action of buffering audio wave data prior to outputting the audio wave data).

Regarding **Claim 18**, in addition to the elements stated above regarding claim 13, Tulkoff further discloses a mixer to digitally mix any number of independent audio streams from any number of applications and sends the new, mixed stream to an audio device (col. 2 lines 63 – 66) (i.e. wherein the audio wave data consumer performs an action of effects-processing the audio wave data prior to outputting the audio wave data).

Regarding **Claim 19**, in addition to the elements stated above regarding claim 13, Tulkoff further discloses that the data being played is copied to a buffer allotted to the client in the mixer (col. 4 lines 24 –25) (i.e. wherein the audio wave data consumer is a data buffer that buffers one or more of the streams of audio wave data), and said

mixer is to digitally mix any number of independent audio streams from any number of applications and sends the new, mixed stream to an audio device (col. 2 lines 63 – 66) (i.e. effects-processes the buffered audio wave data).

Regarding **Claim 20**, in addition to the elements stated above regarding claim 13, Tulkoff further discloses that the audio streams are of varying formats from at least one application program (col. 2 lines 4 – 6) (i.e. wherein the sources are software components).

Regarding **Claim 21**, in addition to the elements stated above regarding claim 13, Tulkoff further discloses a system for producing a single audio stream from a plurality of streams and at least one application program producing the plurality of streams (col. 2 lines 13 – 16) and the mixer acts as an initializer and a handler (col. 2 lines 43 – 44) (i.e. programming objects having interfaces that are callable by a software component to generate the one or more streams of audio wave data).

Regarding **Claim 36**, Tulkoff discloses multiple audio processes that provide multiple streams of audio, a Mixer and an audio device that receive multiple audio streams from audio processes (Fig. 1 elements 10, 12 and 14) (i.e. a plurality of logical bus objects configured to receive audio wave data wherein each logical bus object corresponds to an audio wave data consumer and wherein one or more streams of audio wave data are assigned to a logical bus object based on the function of an audio wave data consumer that corresponds to the logical bus object) and the internal client structures suitably contain all of the audio stream parameters that were set in each audio process (col. 3 lines 19 – 21) (i.e. a data structure), the audio processes make

connections to the mixer (col. 3 lines 17 – 18) and audio streams are sent from the audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12) (i.e. correlating the logical bus object according to a function of an audio wave data consumer that corresponds to a logical bus object). Tulkoff does not disclose wherein each logical bus object is dynamically allocated in response to a need associated with receiving the audio wave data, and wherein at least one of the logical bus objects can be dynamically de-allocated when no longer needed to route a stream of audio wave data. Solomon discloses a system with multiple buses, when a sound card is granted access to a bus, the bus locks out all other agents and all other buses until the transaction for which the sound card was granted access has been entirely completed (col. 3 lines 25 – 38) (i.e. wherein each logical bus object is dynamically allocated in response to a need associated with receiving the audio wave data, and wherein at least one of the logical bus objects can be dynamically de-allocated when no longer needed to route a stream of audio wave data). It would have been obvious to one of ordinary skill in the art at the time of the invention to use Solomon's bus allocation method on Tulkoff's mixer to control the busses used by Tulkoff's audio processes. One would have been motivated to do so to ensure the audio device would be guaranteed allocation of the bus for the time it is needed. See Solomon col. 1 lines 1 – 45.

Regarding **Claim 37**, in addition to the elements listed above regarding claim 36, Tulkoff further discloses multiple audio processes that provide multiple streams of audio, a Mixer and an audio device that receive multiple audio streams from audio

processes (Fig. 1 elements 10, 12 and 14) (i.e. wherein a logical bus object receives one or more of the assigned audio wave data streams and routes the audio wave data streams to the corresponding audio wave data consumer).

Regarding Claim 41 in addition to the elements stated above regarding claim 36, Tulkoff further discloses a single mixer (audio wave data consumer) that receives audio streams from an audio process and each process sends a single stream to the mixer (Fig. 1 elements 10 and 12) (i.e. wherein each logical bus corresponds to a single audio wave data consumer).

Regarding **Claim 42**, in addition to the elements stated above regarding claim 36, Tulkoff further discloses that multiple audio streams from audio processes are sent to a single mixer (Fig. 1 elements 10 and 12) (i.e. wherein at least two of the logical buses correspond to the same audio wave data consumer).

Regarding **Claim 43** in addition to the elements stated above regarding claim 36, Tulkoff further discloses the audio processes make connections to the mixer (col. 3 lines 17 – 18) and audio streams are sent from the audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12) (i.e. wherein a plurality of audio wave data streams are assigned to at least one of the logical buses).

Regarding **Claim 44**, Tulkoff further discloses the mixer keeps track of all of the clients, i.e., audio processes that have made connections to it and the internal structures keep audio stream parameters that were set in the audio process (col. 3 lines 16 – 21) and digitally mixing any number of independent audio streams from any number of applications (col. 2 lines 63 – 66) (i.e. a bus identifier parameter to uniquely

identify a logical bus that corresponds to an audio wave data consumer and a function identifier parameter to identify an effects-processing function of the audio wave data consumer), when a request is received a determination of whether the request is from a new client is made (col. 4 lines 11 – 13) (i.e. a programming reference to identify the audio wave data consumer), and the audio processes make connections to the mixer (col. 3 lines 17 – 18) and audio streams are sent from the audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12) (i.e. wherein at least one stream of audio wave data is routed to a plurality of different logical buses, with the bus identifier parameter being defined according to the function identifier parameter of the corresponding audio wave data consumer).

Regarding **Claim 45**, Tulkoff discloses a Mixer that receives multiple audio streams from multiple audio processes and transfers them to an audio device (Fig. 1 elements 10, 12 and 14) (i.e. providing an audio wave data generation component configured to receive audio content and an instruction to generate one or more streams of audio wave data and providing an audio wave data consumer component configured to receive the one or more streams of audio wave data and providing a logical bus component configured to route the one or more streams of audio wave data to the audio wave data consumer component). Tulkoff does not disclose dynamically providing the logical bus in response to a need associated with receiving the streams of audio wave data or dynamically de-allocating at least one of the logical buses when no longer needed to route a stream of audio wave data. Solomon discloses a system with multiple buses, when a sound card is granted access to a bus, the bus locks out all

other agents and all other buses until the transaction for which the sound card was granted access has been entirely completed (col. 3 lines 25 – 38) (i.e. dynamically providing the logical bus in response to a need associated with receiving the streams of audio wave data or dynamically de-allocating at least one of the logical buses when no longer needed to route a stream of audio wave data). It would have been obvious to one of ordinary skill in the art at the time of the invention to use Solomon's bus allocation method on Tulkoff's mixer to control the busses used by Tulkoff's audio processes. One would have been motivated to do so to ensure the audio device would be guaranteed allocation of the bus for the time it is needed. See Solomon col. 1 lines 1 – 45.

Regarding **Claim 47**, in addition to the elements stated above regarding claim 45, Tulkoff further discloses that the data being played is copied to a buffer allotted to the client in the mixer (col. 4 lines 24 –25) (i.e. wherein the audio wave data consumer component is a data buffer that performs an action of buffering audio wave data).

Regarding **Claim 48**, in addition to the elements stated above regarding claim 45, Tulkoff further discloses a mixer to digitally mix any number of independent audio streams from any number of applications and sends the new, mixed stream to an audio device (col. 2 lines 63 – 66) (i.e. wherein the audio wave data consumer component performs an action of effects-processing the audio wave data).

Regarding **Claim 50**, in addition to the elements stated above regarding claim 45, Tulkoff further discloses a Mixer that receives multiple audio streams from multiple audio processes and an audio device (Fig. 1 elements 10, 12 and 14), the audio

processes make connections to the mixer (col. 3 lines 17 – 18) audio streams are sent from the audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12) (i.e. assigning one or more of the streams of audio wave data to the logical bus component).

Regarding **Claim 51**, in addition to the elements stated above regarding claim 45, Tulkoff further discloses providing a running process in a computer system that allows simultaneous use of an output audio device (col. 2 lines 1 – 3) (i.e. one or more computer executable instructions that, when executed, direct a computing system to perform the method of claim 45).

Regarding **Claim 52**, Tulkoff discloses a Mixer that receives multiple audio streams from multiple audio processes and an audio device (Fig. 1 elements 10, 12 and 14) (i.e. receiving multiple streams of audio wave data and an audio wave data consumer), the audio processes make connections to the mixer (col. 3 lines 17 – 18) (i.e. defining logical buses that each correspond to an audio wave data consumer), that the internal client structures suitably contain all of the audio stream parameters that were set in each audio process (col. 3 lines 19 – 21) (i.e. creating a data structure), the audio processes make connections to the mixer (col. 3 lines 17 – 18) and audio streams are sent from the audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12) (i.e. correlating the logical busses with corresponding audio wave data consumers), audio streams are sent from the audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12). It is inherent that if the streams are sent from the audio process to the mixer that some sort

of connection device or bus must be present to transport the streams (i.e. assigning at least one of the multiple streams of audio wave data to a plurality of the logical buses; and routing any audio wave data stream assigned to a particular logical bus to the audio wave data consumer corresponding to said particular logical bus). Tulkoff does not disclose dynamically defining logical buses in response to a need associated with receiving the streams of audio wave data or dynamically de-allocating at least one of the logical buses when no longer needed. Solomon discloses a system with multiple buses, when a sound card is granted access to a bus, the bus locks out all other agents and all other buses until the transaction for which the sound card was granted access has been entirely completed (col. 3 lines 25 – 38) (i.e. dynamically defining logical buses in response to a need associated with receiving the streams of audio wave data or dynamically de-allocating at least one of the logical buses when no longer needed). It would have been obvious to one of ordinary skill in the art at the time of the invention to use Solomon's bus allocation method on Tulkoff's mixer to control the busses used by Tulkoff's audio processes. One would have been motivated to do so to ensure the audio device would be guaranteed allocation of the bus for the time it is needed. See Solomon col. 1 lines 1 – 45.

Regarding **Claim 53** in addition to the elements stated above regarding claim 52, Tulkoff further discloses the audio processes make connections to the mixer (col. 3 lines 17 – 18) and audio streams are sent from the audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12) (i.e. wherein a plurality of audio wave data streams are assigned to at least one of the logical buses).

Regarding **Claim 54** in addition to the elements stated above regarding claim 52, Tulkoff further discloses a single mixer (audio wave data consumer) that receives audio streams from an audio process and each process sends a single stream to the mixer (Fig. 1 elements 10 and 12) (i.e. wherein each logical bus corresponds to a single audio wave data consumer).

Regarding **Claim 55**, in addition to the elements stated above regarding claim 52, Tulkoff further discloses that multiple audio streams from audio processes are sent to a single mixer (Fig. 1 elements 10 and 12) (i.e. wherein at least two of the logical buses correspond to the same audio wave data consumer).

Regarding **Claim 56**, in addition to the elements stated above regarding claim 52, Tulkoff further discloses providing a running process in a computer system that allows simultaneous use of an output audio device (col. 2 lines 1 – 3) (i.e. computer executable instructions that, when executed, direct a computing system to perform the method of claim 52).

3. **Claims 2, 22 – 30, 32 - 35, 38, 40, 46 and 49** are rejected under 35 U.S.C. 103(a) as being unpatentable over Tulkoff (U.S. Patent 5,890,017) in view of Solomon (U.S. Patent 5,890,017) in further view of Gulick (U.S. Patent 5,717,154).

Regarding **Claim 2**, in addition to the elements stated above regarding claim 1, Gulick discloses that a synthesizer address generator generates a request for two wavetable data samples each frame for each active voice (col. 6 lines 46 – 48) (i.e. generating the streams of audio wave data in response to receiving a synthesizer instruction). Tulkoff discloses that in accordance with the present invention the audio

processes represent sources of audio streams from one or more applications (col. 2 line68 and col. 3 lines 1 – 2). One of ordinary skill in the art at the time of the invention would have been motivated to combine Gulick's synthesizer with the Tulkoff's Solomon combination to provide one or more audio streams for the mixer. Gulick's synthesizer provides a suitable audio stream and therefore it would have been obvious to one of ordinary skill in the art to combine it with Tulkoff's mixer. Gulick discloses that often users desire output of more than one stream at a time. For example, a game may include a background audio stream, while interjecting momentary reaction audio streams according to activity occurring during the playing of the game (col. 1 lines 19 – 29).

Regarding **Claim 22**, in addition to the elements stated above regarding claim 13, Tulkoff further discloses multiple audio processes that provide multiple streams of audio (Fig. 1 element 10) (i.e. one or more audio processes that generate one or more streams of audio data). Tulkoff does not discloses a synthesizer. Gulick discloses that a synthesizer address generator generates a request for two wavetable data samples each frame for each active voice (col. 6 lines 46 – 48) (i.e. one or more synthesizers). The motivation to combine these elements is given above regarding claim 2.

Regarding **Claim 23**, in addition to the elements stated above regarding claim 13, Tulkoff further discloses multiple audio processes that provide multiple streams of audio (Fig. 1 element 10) (i.e. a plurality of audio processes that generate one or more streams of audio wave data). Tulkoff does not discloses a synthesizer. Gulick discloses that a synthesizer address generator generates a request for two wavetable

data samples each frame for each active voice (col. 6 lines 46 – 48) (i.e. one or more synthesizers). The motivation to combine these elements is given above regarding claim 2. Furthermore neither Tulkoff nor Gulick disclose wherein at least one of the synthesizers generates a plurality of outputs, and wherein respective ones of the outputs are provided to different respective logical buses. However, taking the output of Gulick's synthesizer and splitting it to create multiple outputs and attaching them to the mixer would have been obvious to one of ordinary skill in the art at the time of the invention. It is notoriously well known in the art to take an output and split it into multiple outputs. One would be motivated to do so to create a surround sound type or stereo effect. Additionally connecting the multiple outputs to the mixer would have been an obvious variation. The mixer accepts multiple outputs and the synthesizer with the split outputs would have provided this accordingly.

Regarding **Claim 24**, in addition to the elements stated above regarding claim 13, Tulkoff further discloses multiple audio processes that provide multiple streams of audio (Fig. 1 element 10) (i.e. a plurality of audio processes that generate one or more streams of audio wave data). Tulkoff does not discloses a synthesizer. Gulick discloses that a synthesizer address generator generates a request for two wavetable data samples each frame for each active voice (col. 6 lines 46 – 48) (i.e. one or more synthesizers). The motivation to combine these elements is given above regarding claim 2.

Regarding **Claim 25**, Tulkoff discloses multiple audio processes that provide multiple streams of audio, a Mixer and an audio device that receive multiple audio

streams from audio processes (Fig. 1 elements 10, 12 and 14) (i.e. multiple streams of audio wave data and a plurality of audio wave data consumers that receive the multiple streams of audio wave data) and a running process on a computer a method that includes sending a plurality of audio streams of varying formats from at least one application program running on the computer system to the output audio device and the method further includes intercepting the plurality of audio streams with a mixer daemon for maintaining transparency to the at least one application program (col. 2 lines 1 – 9) (i.e. a software component configured to receive one or more of the streams of audio wave data at each of the defined logical buses, and route any audio wave data that is received at a particular logical bus to an audio wave data consumer corresponding to said particular logical bus). Tulkoff does not disclose a synthesizer. Gulick discloses that a synthesizer address generator generates a request for two wavetable data samples each frame for each active voice (col. 6 lines 46 – 48) (i.e. a synthesizer that generates a stream of audio wave data and wherein the synthesizer is configured to route at least one of the streams of audio wave data to different ones of the logical buses). The motivation to combine these elements is given above regarding claim 2.

Regarding **Claim 26**, in addition to the elements stated above regarding claim 25, Tulkoff further discloses multiple audio processes that provide multiple streams of audio, a Mixer and an audio device that receive multiple audio streams from audio processes (Fig. 1 elements 10, 12 and 14) (i.e. wherein a second logical bus is configured to correspond to the audio wave data consumer, receive one or more

additional streams of audio wave data, and route the one or more additional streams of audio wave data to the audio wave data consumer).

Regarding **Claim 27**, in addition to the elements stated above regarding claim 25, Tulkoff discloses audio streams are sent from the audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12) (i.e. routing the stream of audio wave data to the individual logical bus). Tulkoff does not disclose a synthesizer that has a channel that generates a stream of audio wave data. Gulick discloses that the synthesizer generates sounds in response to the wavetable data samples (col. 5 lines 4 – 5) (i.e. wherein the synthesizer has a channel that generates a stream of audio wave data). The motivation to combine these elements is given above regarding claim 2. Neither Tulkoff nor Gulick disclose wherein the software component is configured to define the logical buses dynamically in response to a need associated with receiving the streams of audio wave data, and is further configured to dynamically de-allocate at least one of the logical buses when no longer needed. Solomon discloses a system with multiple buses, when a sound card is granted access to a bus, the bus locks out all other agents and all other buses until the transaction for which the sound card was granted access has been entirely completed (col. 3 lines 25 – 38) (i.e. disclose wherein the software component is configured to define the logical buses dynamically in response to a need associated with receiving the streams of audio wave data, and is further configured to dynamically de-allocate at least one of the logical buses when no longer needed). It would have been obvious to one of ordinary skill in the art at the time of the invention to use Solomon's bus allocation method on Tulkoff's

mixer to control the busses used by Tulkoff's audio processes. One would have been motivated to do so to ensure the audio device would be guaranteed allocation of the bus for the time it is needed. See Solomon col. 1 lines 1 – 45.

Regarding **Claim 28**, in addition to the elements stated above regarding claim 25, Tulkoff discloses audio streams are sent from the audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12), a Mixer and an audio device that receive multiple audio streams from audio processes (Fig. 1 elements 10, 12 and 14) (i.e. route the stream of audio wave data to a plurality of logical buses and wherein the logical buses receive the stream of audio wave data and route the stream of audio wave data to a plurality of corresponding audio wave data consumers). Tulkoff does not disclose a synthesizer that has a channel that generates a stream of audio wave data. Gulick discloses that the synthesizer generates sounds in response to the wavetable data samples (col. 5 lines 4 – 5). (i.e. wherein the synthesizer has a channel that generates a stream of audio wave data). The motivation to combine these elements is given above regarding claim 2.

Regarding **Claim 29**, in addition to the elements stated above regarding claim 25, Tulkoff discloses audio streams are sent from the audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12), a Mixer and an audio device that receive multiple audio streams from audio processes (Fig. 1 elements 10, 12 and 14) (i.e. route the stream of audio wave data to a plurality of logical buses and wherein the logical buses receive the stream of audio wave data and route the stream of audio wave data to a plurality of corresponding audio wave data consumers).

Tulkoff does not disclose a synthesizer that has a channel that generates a stream of audio wave data. Gulick discloses that the synthesizer generates sounds in response to the wavetable data samples (col. 5 lines 4 – 5). It is inherent that placing multiple synthesizers in place of Tulkoff's audio processes will create a plurality of generated streams. (i.e. wherein the synthesizer has a plurality of channels that each generate a stream of audio wave data). The motivation to combine these elements is given above regarding claim 2.

Regarding **Claim 30**, in addition to the elements stated above regarding claim 25, Gulick further discloses that the synthesizer generates sounds in response to the wavetable data samples (col. 5 lines 4 – 5) and synthesizer address generator generates a request for two wavetable data samples each frame for each active voice (col. 6 lines 46 – 49) (i.e. wherein the synthesizer generates a stream of audio wave data in response to a synthesizer instruction).

Regarding **Claim 32**, in addition to the elements stated above regarding claim 25, Tulkoff discloses audio streams are sent from multiple audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12) (i.e. wherein the individual logical bus is configured to receive one or more of the additional streams of audio wave data and route the additional streams of audio wave data to the audio wave data consumer). Tulkoff does not disclose multiple synthesizers. Gulick further discloses that the synthesizer generates sounds in response to the wavetable data samples (col. 5 lines 4 – 5). Replacing Tulkoff's multiple audio processes with Gulick's synthesizer would create multiple synthesizers (i.e. a second synthesizer to generate

additional streams of audio wave data). The motivation to combine these elements is given above regarding claim 2.

Regarding **Claim 33**, in addition to the elements stated above regarding claim 25, Tulkoff discloses audio streams are sent from multiple audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12). This creates multiple buses (i.e. wherein a second logical bus is configured to correspond to the audio wave data consumer, receive one or more of the additional streams of audio wave data and route the additional streams of audio wave data to the audio wave data consumer). Tulkoff does not disclose multiple synthesizers. Gulick further discloses that the synthesizer generates sounds in response to the wavetable data samples (col. 5 lines 4 – 5). Replacing Tulkoff's multiple audio processes with Gulick's synthesizer would create multiple synthesizers (i.e. a second synthesizer to generate additional streams of audio wave data). The motivation to combine these elements is given above regarding claim 2.

Regarding **Claim 34**, in addition to the elements stated above regarding claim 25, Tulkoff further discloses that the internal client structures suitably contain all of the audio stream parameters that were set in each audio process (col. 3 lines 19 – 21) (i.e. a data structure), the audio processes make connections to the mixer (col. 3 lines 17 – 18) and audio streams are sent from the audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12) (i.e. correlating the logical busses with corresponding audio wave data consumers).

Regarding **Claim 35**, in addition to the elements stated above regarding claim 25, Tulkoff further discloses that the internal client structures suitably contain all of the audio stream parameters that were set in each audio process (col. 3 lines 19 – 21) (i.e. a data structure), the audio processes make connections to the mixer (col. 3 lines 17 – 18) and audio streams are sent from the audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12) (i.e. correlating the logical busses with corresponding audio wave data consumers and the audio wave data consumer receives streams of audio wave data from the corresponding logical buses).

Regarding **Claim 38** in addition to the elements listed above regarding claim 36, Tulkoff discloses audio streams are sent from multiple audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12) (i.e. wherein at least one of the streams of audio wave data is provided to different respective logical buses. Gulick further discloses that the synthesizer generates sounds in response to the wavetable data samples (col. 5 lines 4 – 5) (i.e. a synthesizer that generates the one or more streams of audio wave data). The motivation to combine these elements is given above regarding claim 2. Also see the rejection of claim 23.

Regarding **Claim 40**, in addition to the elements stated above regarding claim 36, Gulick further discloses that the synthesizer generates sounds in response to the wavetable data samples (col. 5 lines 4 – 5) and synthesizer address generator generates a request for two wavetable data samples each frame for each active voice (col. 6 lines 46 – 49) (i.e. an audio wave data generation object configured to receive

audio content and an instruction to generate the one or more streams of audio wave data).

Regarding **Claim 46**, in addition to the elements stated above regarding claim 45, Regarding Claim 39 in addition to the elements stated above regarding claim 36, Gulick further discloses that the synthesizer generates sounds in response to the wavetable data samples (col. 5 lines 4 – 5) (i.e. wherein the audio wave data generation component is a synthesizer).

Regarding **Claim 49**, in addition to the elements stated above regarding claim 45, Tulkoff further discloses the audio processes make connections to the mixer (col. 3 lines 17 – 18) and audio streams are sent from the audio processes to the audio device by the mixer (col. 3 lines 32 – 35 and Fig. 1 elements 10 and 12) (i.e. assigning a given one of the streams of audio wave data to a plurality of different logical bus components).

4. **Claims 31 and 39** are rejected under 35 U.S.C. 103(a) as being unpatentable over Tulkoff (U.S. Patent 5,890,017) in view Solomon (U.S. Patent 5,768,545) in further view of Gulick (U.S. Patent 5,717,154) and in further view of Hewitt (U.S. Patent 6,100,461).

Regarding **Claim 31**, in addition to the elements stated above regarding claim 25, Gulick further discloses that the synthesizer generates sounds in response to the wavetable data samples (col. 5 lines 4 – 5) and synthesizer address generator generates a request for two wavetable data samples each frame for each active voice (col. 6 lines 46 – 49). Gulick does not disclose a MIDI system. Hewitt discloses that audio peripherals are commonly available as digital audio systems using a standard

MIDI serial communication protocol for performance of audio voice signals and one type of this audio peripheral is a wavetable-type music synthesizer (col. 1 lines 12 – 16) (i.e. wherein the synthesizer generates a stream of audio wave data in response to a MIDI instruction). One of ordinary skill in the art at the time of the invention would have been motivated to use Gulick's wavetable synthesizer as Hewitt's MIDI system to reliably transfer musical data.

Regarding **Claim 39** in addition to the elements stated above regarding claim 36, Gulick further discloses that the synthesizer generates sounds in response to the wavetable data samples (col. 5 lines 4 – 5) and synthesizer address generator generates a request for two wavetable data samples each frame for each active voice (col. 6 lines 46 – 49). Gulick does not disclose a MIDI system. Hewitt discloses that audio peripherals are commonly available as digital audio systems using a standard MIDI serial communication protocol for performance of audio voice signals and one type of this audio peripheral is a wavetable-type music synthesizer (col. 1 lines 12 – 16) (i.e. a synthesizer that generates the one or more streams of audio wave data in response to a MIDI instruction).

Response to Arguments

Applicant's arguments filed 24 March 2005 have been fully considered but they are not persuasive.

Applicant's arguments with respect to claims 1, 2, 27, 28, 44, 45, 46, 49 and 52 have been considered but are moot in view of the new ground(s) of rejection.

Regarding Claims 1, 2, 27, 28, 44, 45, 46, 49 and 52, In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., the audio data "fans – out") are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

Furthermore in regards to Claims 1, 2, 27, 28, 44, 45, 46, 49 and 52, the limitation states assigning at least one of the multiple streams of audio wave data to a plurality of the logical buses. Tulkoff teaches of audio processes that transmit (via an inherent bus type connection) to the mixer. Examiner points to figure 1 which shows the multiple streams, 5 in the example figure. These five streams are created by the audio processes (assigning at least one of the multiple streams of audio wave data to a plurality of logical buses). Contrary to applicant's argument, this reads upon the claimed limitation.

In addition to Claim 28, applicant asserts that Tulkoff and Gulick do not teach or suggest that a synthesizer has a channel that is configurable to route a stream of audio wave data to a plurality of the logical buses. Tulkoff discloses multiple streams from various audio processes. Gulick discloses a synthesizer. It would be obvious to add a synthesizer as an audio process. A synthesizer with multiple outputs is just an obvious variation, splitting the output of a single output synthesizer is well known in the art.

Tulkoff discloses accepting these audio processes and sending them to a mixer. Examiner points to col. 3 lines 1 – 5 where Tulkoff states the audio streams represent sources of audio from one or more applications. A synthesizer with split outputs is a source of audio with one or more applications. As such, the rejection stands.

Applicant's arguments with respect to claims 13, 22 – 25, 36, 38, 39 and 40 have been considered but are moot in view of the new ground(s) of rejection.

Further regarding Claim 25, in addition to what is cited in response to the arguments of claim 1, The Tulkoff and Gulick combination does teach a synthesizer configured to route a stream of audio wave data to different ones of the logical buses. Tulkoff discloses multiple streams from various audio processes. It would be obvious to add a synthesizer as an audio process. A synthesizer with multiple outputs is just an obvious variation, splitting the output of a single output synthesizer is well known in the art. Tulkoff discloses accepting these audio processes and sending them to a mixer. Examiner points to col. 3 lines 1 – 5 where Tulkoff states the audio streams represent sources of audio from one or more applications. A synthesizer with split outputs is a source of audio with one or more applications. As such, the rejection stands.

Conclusion

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Andrew C Flanders whose telephone number is (571) 272-7516. The examiner can normally be reached on M-F 8:30 - 5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Sinh Tran can be reached on (571) 272-7564. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

acf

SINH TRAN
[illegible]ORY PATENT EXAMINER